JPPIPA 11(12) (2025)

Jurnal Penelitian Pendidikan IPA

http://jppipa.unram.ac.id/index.php/jppipa/index

Design and Implementation of a Public Sentiment Prediction Framework on Budget Efficiency Policy using Support Vector Machine, Naïve Bayes, and Random Forest

M Sahyudi[1*], Sampurna Dadi Riskiono[1], Ryan Randy Suryono[1]

[1] Master of Computer Science Study Program, Universitas Teknokrat Indonesia, Bandar Lampung, Indonesia.

Received: October 23, 2025
Revised: November 21, 2025
Accepted: December 25, 2025
Published: December 31, 2025

Corresponding Author:
M Sahyudi
M_sahyudi@teknokrat.ac.id

DOI: 10.29303/jppipa.v11i12.13561

Abstract: Public sentiment toward government budget efficiency policies has become increasingly visible through social media platforms, where citizens actively express opinions, support, and criticism. This study aims to analyze public sentiment toward budget efficiency policies using data collected from the social media platform X (formerly Twitter). A total of 2,000 public comments related to budget efficiency policies were collected through web scraping using the X API. The data were preprocessed through normalization, case folding, text cleaning, tokenization, stopword removal, and stemming. Sentiment classification was conducted using three machine learning algorithms: Naive Bayes, Support Vector Machine (SVM), and Random Forest. Model performance was evaluated using accuracy, precision, recall, and F1-score. The results indicate that SVM achieved the highest accuracy, while Random Forest demonstrated superior recall in identifying positive sentiment. These findings suggest that Random Forest is particularly suitable for sentiment analysis tasks where minimizing false negatives is important, while SVM performs well in overall classification accuracy. This research contributes to the comparative evaluation of machine learning models for public sentiment analysis on policy-related issues using social media data.

Keywords: Budget Efficiency Policy; Naïve Bayes; Random forest; Sentiment; Support vector machine

Introduction

Budget efficiency policy is an important strategy adopted by the government to ensure that the use of state resources can be conducted optimally, efficiently, and sustainably. The latest concrete measure, Presidential Instruction Number 1 of 2025, issued on January 22, 2025, stipulates a reduction in state spending by Rp306.69 trillion. This includes Rp256.1 trillion in cuts from ministries and government institutions as well as Rp50.59 trillion from regional transfers, aimed at reducing non-priority spending and ensuring fiscal sustainability (Kementrian Sekretariat Negara, 2025). In practice, budget efficiency is often achieved through spending cuts in specific sectors and budget reallocation based on national development priorities. The objective is to reduce waste, improve the quality of public spending, and strengthen the economic discipline of the country (Salman & Ikbal, 2025).

The level of public acceptance and perception is influenced by many factors, including how transparent the government is in communicating policy objectives and the extent to which the public directly perceives its impact. A rapidly developing method to explore public opinion is through the use of information technology, particularly data from social media (Pratama, 2025). Through the increasing use of digital platforms, social media has become a very active and open space for public expression (Mas'ud et al., 2025). On this platform, the public is free to express the views on current policy issues, including budget efficiency. The large volume of data, diversity of topics, and real-time nature make social media a rich source of data for analysis (Maharani, 2024).

How to Cite:
Sahyudi, M., Riskiono, S. D., & Suryono, R. R. (2025). Design and Implementation of a Public Sentiment Prediction Framework on Budget Efficiency Policy using Support Vector Machine, Naïve Bayes, and Random Forest. *Jurnal Penelitian Pendidikan IPA*, *11*(12), 1370-1380. https://doi.org/10.29303/jppipa.v11i12.13561

Sentiment analysis is a research method that aims to evaluate and classify emotions or attitudes contained in text using digital media, particularly in the comments area (Hayami et al., 2022). This method focuses on the process of differentiating text based on sentiment content by identifying how a statement contains positive or negative connotations toward a particular subject (Insan et al., 2023). The method is performed by exploring and analyzing unstructured text to understand public opinion about various specific products, topics, or services (Kusnadi et al., 2024). However, sophisticated computational methods are needed to process this large amount of unstructured data, such as Naive Bayes, Support Vector Machine (SVM), and Random Forest (Hafika et al., 2025).

This research aims to compare the performance of three machine learning algorithms, namely Naive Bayes, SVM, and Random Forest, in analyzing the public sentiment prediction framework towards budget efficiency policies. Through a comprehensive evaluation including accuracy, precision, recall, and F1-score, this research aims to identify the most effective algorithm in classifying public sentiment towards budget efficiency policies. The main focus is to systematically compare algorithm performance in analyzing public sentiment patterns, improving the understanding of public views and experiences. It also aims to contribute methodologically to the development of analytical frameworks for predicting public sentiment, particularly in budget efficiency policies.

Method

This study employed a web scraping approach to collect public comments related to budget efficiency policies from the social media platform X (formerly Twitter). Data collection was conducted using the X API by retrieving public posts containing relevant keywords and hashtags associated with budget efficiency policies (Husada & Paramita, 2021). A total of 2,000 comments were successfully collected and used as the dataset for this research. The overall research workflow applied in this study is illustrated in Figure 1.

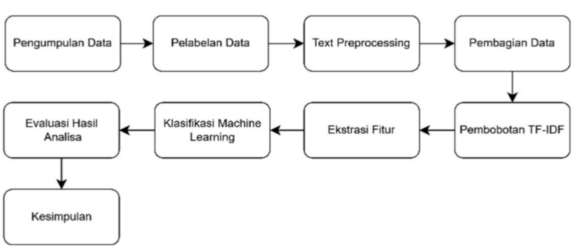

Figure 1. Research Procedures

The collected data underwent several preprocessing stages to improve data quality and ensure suitability for sentiment analysis. These stages included case folding to convert all text to lowercase, text cleaning to remove URLs, punctuation, numbers, and special characters, tokenization to split text into individual words, stopword removal to eliminate non-informative terms, and stemming to reduce words to their root forms. Sentiment labeling was performed by categorizing the comments into three classes: positive, negative, and neutral. The dataset was then divided into training data (80%) and testing data (20%) using a random state of 42 to ensure reproducibility of the results.

Feature extraction was conducted using the Count Vectorizer method to transform textual data into numerical representations based on word frequency. This method represents each document as a vector of term frequencies, enabling machine learning algorithms to process textual data effectively while reducing dimensionality (Vincent et al., 2024). Prior to model training, the labeled dataset consisted of 1,380 positive comments, 532 negative comments, and 88 neutral comments. This distribution indicates a noticeable class imbalance, with neutral sentiment being significantly underrepresented in the dataset, as shown in Table 1.

Table 1. Feature Extraction Results

Label	Amount of Data
Positive	1380
Negative	532
Neutral	88

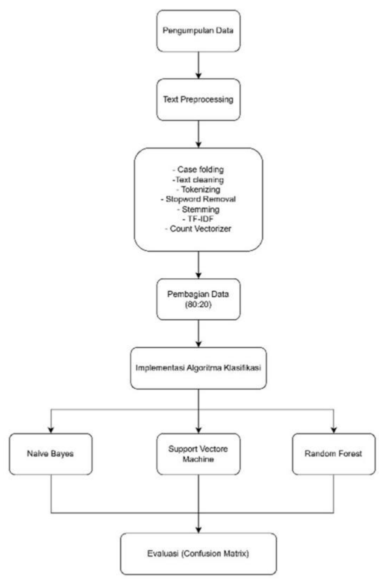

Figure 2. Sentiment Classification Flow

Three supervised machine learning algorithms were applied for sentiment classification: Naive Bayes, Support Vector Machine (SVM), and Random Forest. Naive Bayes was used as a probabilistic baseline model, SVM was applied to identify optimal decision boundaries between sentiment classes, and Random Forest utilized an ensemble of decision trees to improve classification robustness. Model performance was evaluated using a confusion matrix and standard classification metrics, including accuracy, precision, recall, and F1-score. The following is the Sentiment Analysis Classification Flow, as shown in the Figure 2.

The sentiment analysis classification flow applied in this study is illustrated in Figure 2. This figure summarizes the main stages of the research process, including data collection, text preprocessing, feature representation, data splitting, sentiment classification using Naive Bayes, Support Vector Machine (SVM), and Random Forest, as well as model evaluation using a confusion matrix and standard performance metrics.

Result and Discussion

System Design

During this stage, a design and several other inputs and outputs were produced. These included documents containing designs, patterns, databases, and components needed to create the system.

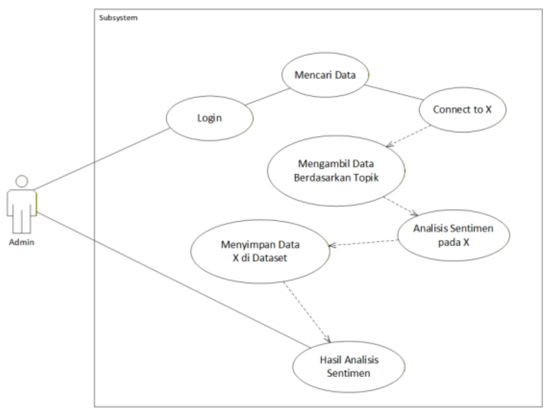

Figure 3. Use Case Diagram

A use case diagram described the interaction between the system and the actor. The process also described the type of interaction between the system user and the system. In Figure 1, the Use Case Diagram included a single actor, namely the user, who interacted with the framework designed to predicting public sentiment toward budget efficiency policies using Naïve Bayes, SVM, and Random Forest (Bei & Sudin, 2021).

Entering the main application of the public sentiment prediction framework design and implementation system for budget efficiency policies, the user first logged in by entering the appropriate username and password. The application crawled/searched for data/comments in X using the API provided by X. The application of predictive framework analysis of public sentiment towards budget efficiency policies retrieved comment data based on topic ID requests. The analysis system for predicting public sentiment towards budget efficiency policies analyzed comments and classified comment data. Comment data was stored in the database. The application of the public sentiment prediction framework to budget efficiency policies produced a sentiment report (Amaliah et al., 2022).

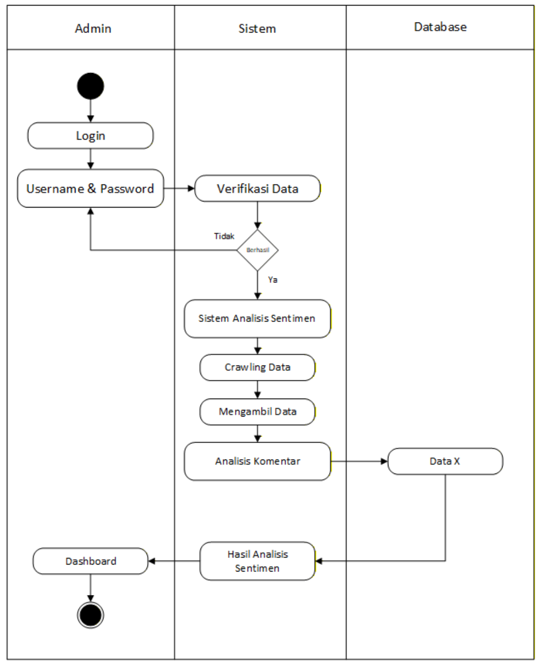

Figure 4. Activity Diagram

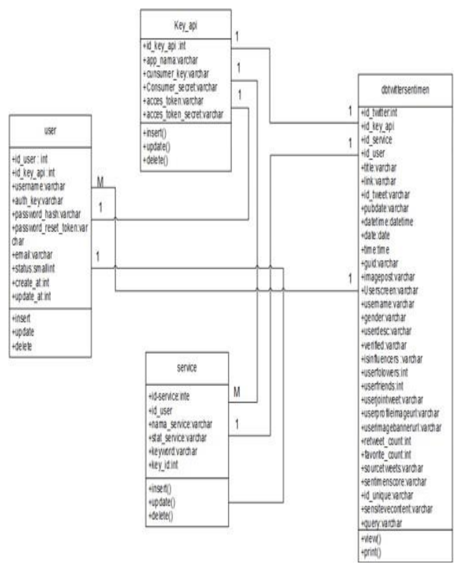

Figure 5. Class Diagram

An activity diagram design represented the flow of activities or workflow in a system that was performed. The diagram was also used to define or group the visual flow of the system.

The design of a class diagram was a picture or representation of the interactions occurring between the system and its environment, which provided an overview of the system. It also showed the relationships in the design system and the implementation system of the public sentiment prediction framework for budget efficiency policies (Jasmarizal et al., 2024).

Input and Output Design

The application was called the Sentiment Analysis Application (SAP), where a web-based SAP connected directly to collect data from one account for testing. This was then used to measure public sentiment predictions concerning budget efficiency policies (Kaharudin & Supriyadi, 2023). The following discussion presented and explained the results of the web-based sentiment analysis application design.

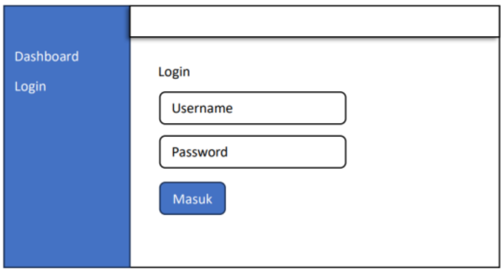

Figure 6. Login Page

A website login page was a dedicated page where users logged into the system by entering a username and password to verify identity, as well as access personalized features or restricted areas. This page served as a security gateway to protect data and ensure that only authorized users could access the content, as well as the functionality behind the page.

Figure 7. Home Page

The home page was the first page visitors encountered when accessing a website and served as the front door, introducing the site. It provided an overview of the content of the site, showed links to other main pages, and served as the starting point for user navigation.

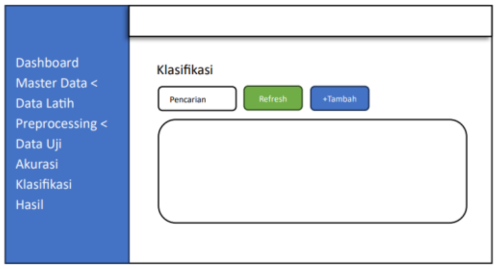

Figure 8. Classification Page

Classification was a method in supervised machine learning in which a model was trained to predict the appropriate discrete labels or categories for new input data, based on patterns learned from the labeled training data. The primary objective of classification was to systematically group observations or data points into distinct classes or categories.

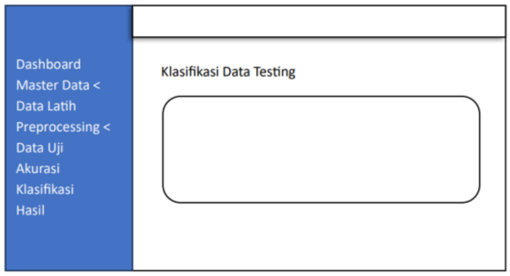

Figure 9. Testing Data Classification Page

A data testing classification page signified a section or interface in a system, or application that was used to evaluate model performance using test data, particularly in the context of machine learning or data mining. This page served as a bridge between the finished model and real-world data, allowing users or developers to assess the effectiveness of the model in making accurate predictions.

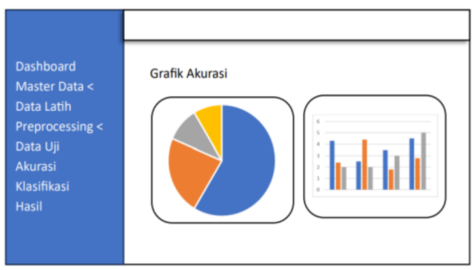

Figure 10. Accuracy Graph Page

An accuracy graph page was an interface, web page, piece of software, or report that showed data visualizations in graphical form. This signified the accuracy and correctness of a model, system, or measurement result. The page served as a visual dashboard that presented important information about how correctly or precisely a system would be performing.

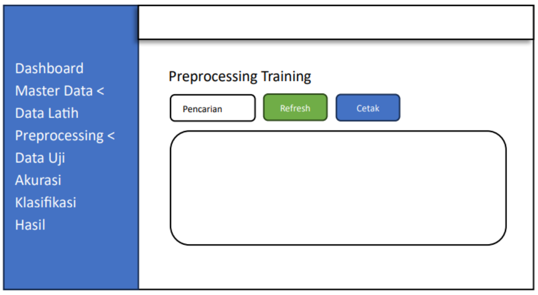

Figure 11. Training Preprocessing Page

The training preprocessing page referred to a section or module in an application, software, or project (particularly in the context of machine learning or data mining) dedicated to preprocessing training data. The page was the process of preparing and transforming raw data used to train a machine learning model. The objective was to clean, organize, and format the data, making it more useful as well as suitable for the algorithms applied (Muhammadin & Sobari, 2021; Normawati & Prayogi, 2021).

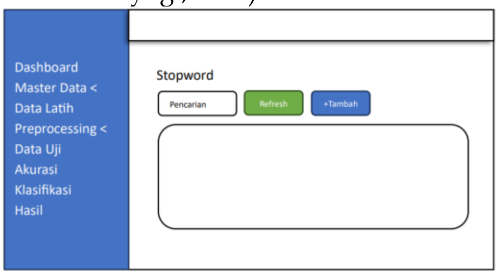

Figure 12. Stopword Page

A stopword page was a list of common words that were ignored or removed during text processing because the words were deemed not to contribute significantly to the meaning or main information in a sentence. The term stopword page possibly referred to a list (or stop list) of these words.

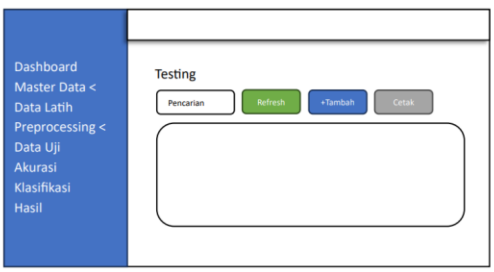

Figure 13. Testing Page

The testing phase referred to the testing data or phase during the process. This was a crucial component of the ML model development cycle, serving to evaluate the performance of the trained model. The testing phase in machine learning was a crucial phase for validating the quality and readiness of an AI model for use in real-world applications.

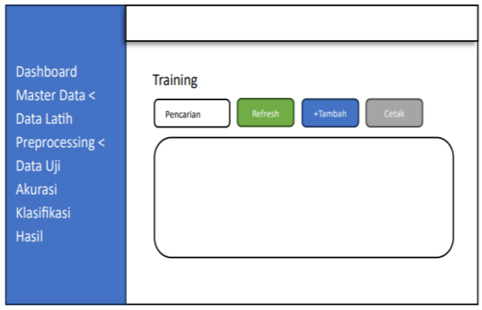

Figure 14. Training Page

The training phase in machine learning was a crucial initial phase where the ML model was exposed to a large amount of labeled data. The primary objective of this phase was enabling the model to learn patterns and relationships in the input data, allowing the model to generate accurate predictions or decisions for new, previously unobserved data.

System Implementation

System implementation was the process of putting a designed system into operation and actual use to achieve specific objectives. This phase included building, testing, installing, training users, and converting to the new system, as well as ensuring the system met stakeholder needs.

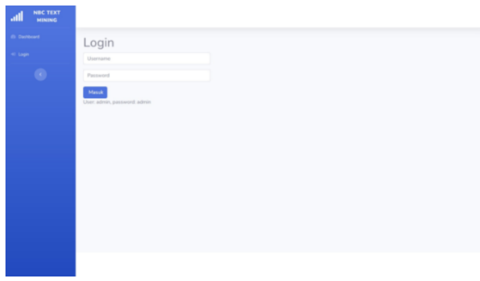

Figure 15. Login Page

A website login page was a dedicated page where users logged into the system by entering a username and password to verify identity as well as access personalized features or restricted areas. This page served as a security gateway to protect data and ensure that only authorized users could access the content and functionality behind the page.

The home page was the first page visitors encountered when accessing a website and served as the front door, introducing the site. It provided an overview of the content of the site, showed links to other main pages, and served as the starting point for user navigation.

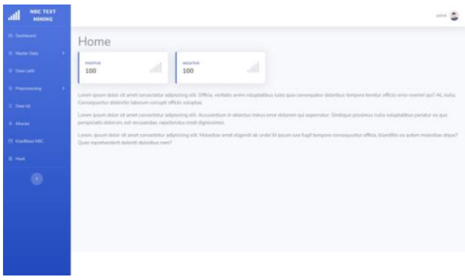

Figure 16. Home Page

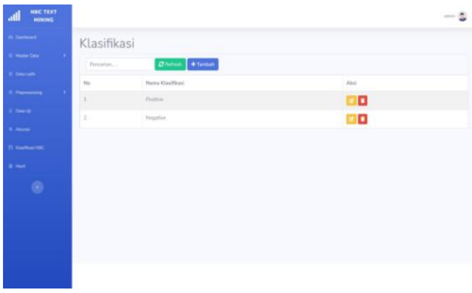

Figure 17. Classification Page

Classification was a method in supervised machine learning in which a model was trained to predict the appropriate discrete labels or categories for new input data, based on patterns learned from the labeled training data. The primary objective of classification was to systematically group observations and data points into distinct classes or categories.

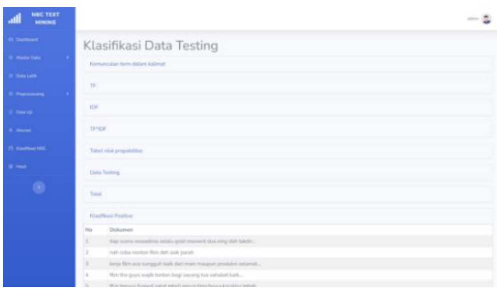

Figure 18. Testing Data Classification Page

A data testing classification page signified a section or interface in a system, or application that was dedicated to evaluating model performance using test data, particularly in machine learning or data mining. This page served as a bridge between the finished model and real-world data, allowing users or developers to assess the effectiveness of the model in making accurate predictions.

An accuracy graph page was an interface, web page, piece of software, or report that showed data visualizations in graphical form. This signified the accuracy and correctness of a model, system, or measurement result. The page served as a visual dashboard that presented important information about how correctly or precisely a system was performing.

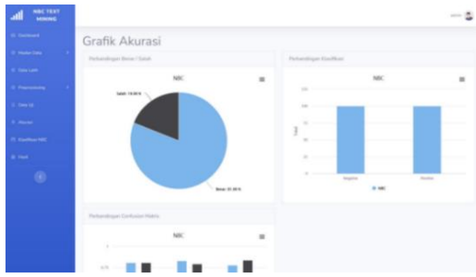

Figure 19. Accuracy Graph Page

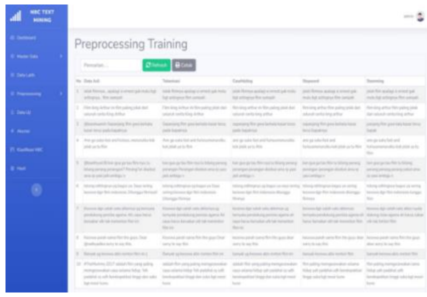

Figure 20. Training Preprocessing Page

The training preprocessing page referred to a section or module in an application, software, or project (particularly in the context of machine learning or data mining) dedicated to preprocessing training data. The phase was the process of preparing and transforming raw data used to train a machine learning model. The goal was to clean, organize, and format the data to make it more useful and suitable for the algorithms to be used.

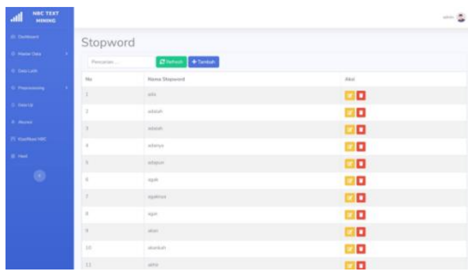

Figure 21. Stopword Page

A stopword page was a list of common words that were ignored or removed during text processing because the words were deemed not to contribute significantly to the meaning or main information in a sentence (Saputra et al., 2021). The term stopword page possibly referred to a list (or stop list) of such words.

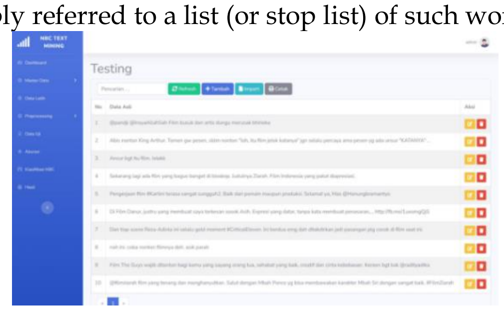

Figure 22. Testing Page

The testing phase referred to the testing data or phase during the process. This was a crucial component of the ML model development cycle, serving to evaluate the performance of the trained model. The testing phase in machine learning was a crucial phase for validating the quality and readiness of an AI model for use in real-world applications.

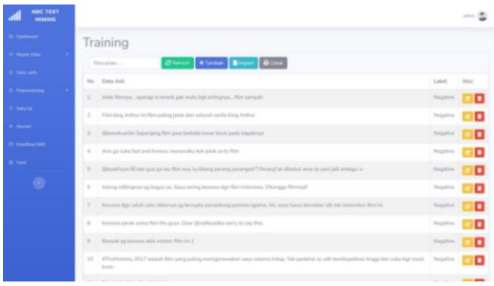

Figure 23. Training Page

The training phase in machine learning was a crucial initial phase where the ML model was exposed to a large amount of labeled data. The primary objective of this phase was to enable the model to recognize patterns and relationships in the input data, allowing accurate predictions or decisions about new, previously unobserved data.

Naive Bayes Classification Results

The performance of the Naive Bayes classifier was evaluated using the test dataset (Suryati et al., 2023), which constituted 20% of the total collected data. As presented in Table 2, the model achieved an overall accuracy of 85% on the test data.

Table 2. Naive Bayes Classification Results

Labels	Precision	Recall	F1-Score	Support
Negative	0.74	0.86	0.79	104
Neutral	0.00	0.00	0.00	21
Positive	0.91	0.92	0.91	275
Accuracy			0.85	400
Macro avg	0.55	0.59	0.57	400
Weighted avg	0.81	0.85	0.83	400

For the positive sentiment class, Naive Bayes demonstrated strong performance, achieving a precision of 91%, a recall of 92%, and an F1-score of 91%. These results indicate that the model was effective in identifying positive sentiment within the test dataset. For the negative class, the model achieved a precision of 74% and a recall of 86%, suggesting a reasonable ability to distinguish negative sentiment.

However, the model failed to effectively classify the neutral sentiment class, as reflected by precision, recall, and F1-score values of 0. This limitation can be attributed to the relatively small number of neutral samples in the dataset, which resulted in class imbalance and reduced the model's ability to learn representative patterns for the neutral class. This issue is also reflected in the lower macro-average scores compared to the weighted-average scores.

Overall, the Naive Bayes classifier showed satisfactory performance on the test data, particularly in classifying positive sentiment, while its performance on minority classes highlights the impact of class imbalance on multi-class sentiment classification.

Classification Results SVM

The performance of the Support Vector Machine (SVM) classifier was evaluated using the test dataset (Moraes et al., 2013; Zhu et al., 2010), which represented 20% of the total collected comments. As shown in Table 3, the SVM model achieved an overall accuracy of 83% on the test data.

Table 3. SVM Classification Results

Labels	Precision	Recall	F1-Score	Support
Negative	0.74	0.73	0.74	104
Neutral	0.18	0.10	0.12	21
Positive	0.89	0.93	0.91	275
Accuracy			0.83	400
Macro Avg	0.61	0.59	0.59	400
Weighted Avg	0.82	0.83	0.82	400

For the positive sentiment class, the SVM classifier achieved a precision of 89%, a recall of 93%, and an F1-score of 91%, indicating a strong capability in identifying positive sentiment. The model also demonstrated moderate performance in classifying negative sentiment, with a precision of 74% and a recall of 73%.

However, similar to the Naive Bayes model, SVM showed limited performance in classifying the neutral sentiment class. The low precision (18%) and recall (10%) indicate that the model struggled to correctly identify neutral comments, which can be attributed to the small number of neutral samples in the dataset and the resulting class imbalance. This limitation is reflected in the relatively low macro-average scores compared to the weighted-average scores.

Overall, the SVM classifier demonstrated effective performance on the test data, particularly for positive sentiment classification, while its limitations in handling minority classes highlight the impact of class imbalance in multi-class sentiment analysis.

Random Forest Classification Results

The performance of the Random Forest classifier was evaluated using the test dataset (Bafail, 2024; He et al., 2024; Khoiruddin et al., 2022; Zhang, 2024), which accounted for 20% of the total collected comments. As presented in Table 4, the Random Forest model achieved an overall accuracy of 85% on the test data.

Table 4. Random Forest Classification Results

Labels	Precision	Recall	F1-Score	Support
Negative	0.78	0.76	0.77	104
Neutral	0.00	0.00	0.00	21
Positive	0.87	0.95	0.91	275
Accuracy			0.85	400
Macro Avg	0.55	0.57	0.56	400
Weighted Avg	0.80	0.85	0.83	400

For the positive sentiment class, Random Forest demonstrated strong recall performance, achieving a recall value of 95%, indicating its effectiveness in identifying most positive sentiment samples. The model achieved a precision of 87% and an F1-score of 91% for the positive class, reflecting a balanced performance between precision and recall. In addition, the model showed moderate performance in classifying negative sentiment, with a precision of 78% and a recall of 76%.

However, the Random Forest classifier was unable to effectively identify the neutral sentiment class, as indicated by precision, recall, and F1-score values of 0. This limitation is primarily attributable to the small number of neutral samples in the dataset, which led to class imbalance and reduced the model's ability to learn representative patterns for the neutral class. This issue is also reflected in the relatively low macro-average scores.

Overall, the Random Forest model demonstrated reliable performance on the test data, particularly in terms of recall for positive sentiment classification, highlighting its suitability for applications where minimizing false negatives is a priority.

Evaluation

Based on the evaluation of the three classification models Naive Bayes, Support Vector Machine (SVM), and Random Forest, the results demonstrate varying performance across different evaluation metrics on the test dataset, which represented 20% of the total collected comments. In terms of overall accuracy, both Naive Bayes and Random Forest achieved comparable performance, each obtaining an accuracy score of 85%, while SVM recorded a slightly lower accuracy of 83%. These results indicate that all three models performed reasonably well in classifying sentiment on the test data, with no substantial difference in overall accuracy.

Regarding precision for the positive sentiment class, SVM achieved the highest value at 89%, followed by Random Forest at 87% and Naive Bayes at 91%. This indicates that all models were capable of producing reliable positive sentiment predictions, with relatively small differences among them.

In contrast, recall results revealed more distinct differences between the models. Random Forest achieved the highest recall for the positive class at 95%, indicating its strong ability to correctly identify most positive sentiment samples. SVM followed with a recall of 93%, while Naive Bayes achieved a recall of 92%. This suggests that Random Forest is particularly effective in minimizing false negatives for positive sentiment classification.

The weighted-average F1-scores of the three models were relatively similar, ranging from 0.82 to 0.83, reflecting a balanced trade-off between precision and recall across sentiment classes. However, the lower macro-average scores observed across all models indicate challenges in handling the neutral sentiment class, primarily due to class imbalance in the dataset.

Overall, the evaluation results indicate that while all three models demonstrate comparable classification performance, Random Forest shows an advantage in recall for positive sentiment, making it more suitable for applications where identifying positive public sentiment is prioritized.

Discussion

The comparative performance of the three machine learning models Naive Bayes, Support Vector Machine (SVM), and Random Forest is illustrated in Figure 24. The comparison is based on four evaluation metrics: accuracy, precision, recall, and F1-score, using the test dataset.

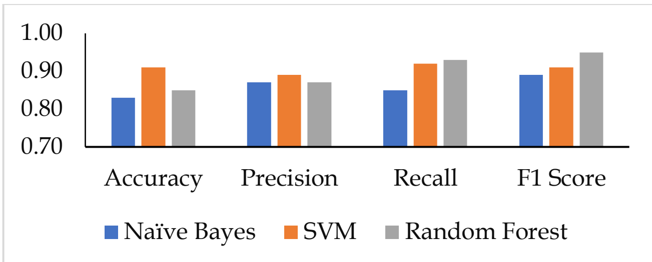

Figure 24. Model Comparison Results

In terms of overall accuracy, Naive Bayes and Random Forest achieved comparable results, each with an accuracy of 0.85, while SVM obtained a slightly lower accuracy of 0.83. These results indicate that all three models demonstrated similar effectiveness in classifying sentiment from public comments related to budget efficiency policies.

Regarding precision for the positive sentiment class, SVM achieved the highest value at 0.89, followed by Random Forest at 0.87 and Naive Bayes at 0.91. This suggests that all models were capable of producing reliable positive sentiment predictions, with relatively small differences among them.

More noticeable differences were observed in recall performance. Random Forest achieved the highest recall value for the positive class at 0.95, indicating its strong ability to correctly identify most positive sentiment samples. SVM followed with a recall of 0.93, while Naive Bayes achieved a recall of 0.92. This result highlights the strength of Random Forest in minimizing false negatives.

The weighted-average F1-scores across the three models were relatively similar, ranging between 0.82 and 0.83, indicating a balanced trade-off between precision and recall. However, the lower macro-average scores observed across all models reflect the difficulty in accurately classifying the neutral sentiment class, which was underrepresented in the dataset.

Overall, the discussion confirms that while no single model overwhelmingly outperformed the others across all metrics, Random Forest demonstrated an advantage in recall for positive sentiment classification, making it particularly suitable for applications where capturing positive public sentiment is a priority.

Conclusions

This study evaluated the performance of Naive Bayes, Support Vector Machine (SVM), and Random Forest for sentiment classification of public comments related to budget efficiency policies collected from platform X. The results indicate that all three models achieved comparable overall accuracy on the test dataset, while exhibiting different strengths across evaluation metrics. Random Forest demonstrated the highest recall for positive sentiment, making it effective in identifying most positive samples, whereas SVM showed strong precision in positive sentiment classification and Naïve Bayes provided competitive performance across metrics. However, all models experienced limitations in classifying neutral sentiment due to class imbalance in the dataset. In addition, this study demonstrates how sentiment analysis models can be integrated into a software based sentiment analysis application, supporting the design and implementation aspects of sentiment analysis systems from a software engineering perspective.

Acknowledgement
The authors are grateful for the contributions of all participants whose scientific work supported this systematic literature review as well as the institution that provided academic databases facilitating access to high-quality scientific publications.

Author Contributions
All authors contributed to the conception and design of the study. Data collection and preprocessing were performed by the authors, including web scraping from platform X and sentiment labeling. The development and implementation of the machine learning models Naive Bayes, Support Vector Machine (SVM), and Random Forest were carried out collaboratively. All authors participated in the analysis and interpretation of the results, as well as in drafting, reviewing, and revising the manuscript. All authors have read and approved the final version of the manuscript.

Funding
This research received no external funding.

Conflict of Interest
The author stated that there was no conflict of interest between the author and the object of research in this paper.

References

Amaliah, S., Nusrang, M., & Aswi, A. (2022). Penerapan metode Random Forest untuk klasifikasi varian minuman kopi di Kedai Kopi Konijiwa Bantaeng. *Journal of Statistics and Its Application on Teaching and Research*, *4*(3), 121–127. https://doi.org/10.35580/variansiunm31

Bafail, O. (2024). Optimizing smart city strategies: A data-driven analysis using random forest and regression analysis. *Applied Sciences*, *14*(23), 11022. https://doi.org/10.3390/app142311022

Bei, F., & Sudin, S. (2021). Analisis sentimen aplikasi tiket online di Play Store menggunakan metode Support Vector Machine (SVM). *Prosiding Seminar Nasional Sistem Informasi Dan Manajemen Informatika Universitas Nusa Putra*, *1*(1), 91–97. Retrieved from https://sismatik.nusaputra.ac.id/index.php/sismatik/en/article/view/13

Hafika, R. A., Hafiz, A., Waruwu, S. A., Noor, M. Y., Sitohang, Y. A. A., & Saputra, K. (2025). Efisiensi anggaran dalam wacana publik: Analisis sentimen platform X dengan Naive Bayes. *Jurnal Mahasiswa Teknik Informatika*, *9*(4), 6093–6099. Retrieved from https://mail.ejournal.itn.ac.id/index.php/jati/article/view/13967

Hayami, R., Soni, & Gunawan, I. (2022). Klasifikasi jamur menggunakan algoritma Naïve Bayes. *Journal Computer Sciences, Informatics, Technology*, *3*(1), 28–33. https://doi.org/10.37859/coscitech.v3i1.3685

He, Y., Wu, C., & Fan, Y. (2024). Exploring the drivers of local government budget coordination: A random forest regression analysis. *International Review of Economics & Finance*, *93*, 1104–1113. https://doi.org/10.1016/j.iref.2024.04.004

Husada, H. C., & Paramita, A. S. (2021). Analisis sentimen pada maskapai penerbangan di platform Twitter menggunakan algoritma Support Vector Machine (SVM. *Teknika*, *10*(1), 18–26. https://doi.org/10.34148/teknika.v10i1.311

Insan, M. K., Hayati, U., & Nurdiawan, O. (2023).

Analisis sentimen aplikasi Brimo pada ulasan pengguna di Google Play menggunakan algoritma Naive Bayes. *Jurnal Teknik Informatika*, *7*(1), 478–483. Retrieved from https://mail.ejournal.itn.ac.id/index.php/jati/article/view/6373

Jasmarizal, J., Junadhi, Rahmaddeni, & Anam, M. K. (2024). Penerapan metode Support Vector Machine untuk analisis sentimen terhadap produk skincare. *The Indonesian Journal of Computer Science*, *13*(1), 1438–1450. https://doi.org/10.33022/ijcs.v13i1.3654

Kaharudin, A., & Supriyadi, A. A. (2023). Analisis sentimen pada media sosial dengan teknik kecerdasan buatan Naïve Bayes: Kajian literatur review. *Jurnal Ilmu Komputer Dan Sains*, *2*(6), 1642–1649. Retrieved from https://shorturl.asia/f8ugN

Kementrian Sekretariat Negara, R. (2025). *Instruksi Presiden (Inpres) Nomor 1 Tahun 2025 tentang Efisiensi Belanja dalam Pelaksanaan Anggaran Pendapatan dan Belanja Negara dan Anggaran Pendapatan dan Belanja Daerah Tahun Anggaran 2025*. Retrieved from https://peraturan.bpk.go.id/Details/313401/inpres-no-1-tahun-2025

Khoiruddin, M., Junaidi, A., & Saputra, W. A. (2022). Klasifikasi Penyakit Daun Padi Menggunakan Convolutional Neural Network. *Journal of Dinda: Data Science, Information Technology, and Data Analytics*, *2*(1), 37–45. Retrieved from https://journal.ittelkom-pwt.ac.id/index.php/dinda/article/view/341

Kusnadi, P. A., Padilah, T. N., & Sari, B. N. (2024). Analisis sentimen penerapan sistem pembayaran tol multi lane free flow menggunakan Naïve Bayes classifier. *Jurnal Teknik Informatika*, *8*(4), 7398–7404. https://doi.org/10.36040/jati.v8i4.10232

Maharani, M. Z. (2024). Analisis sentimen positif terhadap Avoskin sebagai eco friendly brand di media sosial X dan TikTok. *Filosofi: Publikasi Ilmu Komunikasi, Desain, Seni Budaya*, *1*(3), 125–140. https://doi.org/10.62383/filosofi.v1i3.169

Mas'ud, F., Jeluhur, H., Negat, K., Tefa, A., Uly, M., & Amtiran, M. (2025). Etika dalam media sosial antara kebebasan ekspresi dan tanggung jawab digital. *Jimmi: Jurnal Ilmiah Mahasiswa Multidisiplin*, *2*(2), 235–246. https://doi.org/10.71153/jimmi.v2i2.289

Moraes, R., Valiati, J. F., & Neto, W. P. G. (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. *Expert Systems with Applications*, *40*(2), 621–633. https://doi.org/10.1016/j.eswa.2012.07.059

Muhammadin, A., & Sobari, I. A. (2021). Analisis sentimen pada ulasan aplikasi Kredivo dengan algoritma SVM dan NBC. *Reputasi Jurnal Rekayasa Perangkat Lunak*, *2*(2), 85–91. https://doi.org/10.31294/reputasi.v2i2.785

Normawati, D., & Prayogi, S. A. (2021). Implementasi Naïve Bayes classifier dan confusion matrix pada analisis sentimen berbasis teks pada Twitter. *Journal Sains Komputer Informatika*, *5*(2), 697–711. Retrieved from https://shorturl.asia/3L1aR

Pratama, M. M. H. R. (2025). Comparison of Support Vector Machine (SVM) and Naïve Bayes algorithm performance in analyzing Garuda bird design sentiment in IKN. *Journal of Informatics and Telecommunication Engineering*, *8*(3), 1–8. https://doi.org/10.31289/jite.v8i3Spc.14830

Salman, & Ikbal, M. (2025). Analisis efektivitas kebijakan efisiensi anggaran: Ditinjau dari aspek ekonomi. *Journal of Economics Development Research*, *1*(2), 68–72. https://doi.org/10.71094/joeder.v1i2.99

Saputra, I., Pambudi, R. S. A., Darono, H. E., Amsury, F., Fahdia, M. R., Ramadhan, B., & Ardiansyah, A. (2021). Analisis sentimen pengguna marketplace Bukalapak dan Tokopedia di Twitter menggunakan machine learning. *Faktor Exacta*, *13*(4). https://doi.org/10.30998/faktorexacta.v13i4.7074

Suryati, E., Styawati, & Aldino, A. A. (2023). Analisis sentimen transportasi online menggunakan ekstraksi fitur model Word2Vec text embedding dan algoritma Support Vector Machine (SVM. *Jurnal Teknologi Dan Sistem Informatika*, *4*(1), 96–106. Retrieved from https://shorturl.asia/gaUo5

Vincent, R., Maulana, I., & Komarudin, O. (2024). Perbandingan klasifikasi Naive Bayes dan Support Vector Machine dalam analisis sentimen dengan multiclass di Twitter. *Jurnal Teknik Informatika*, *7*(4), 2496–2505. Retrieved from https://mail.ejournal.itn.ac.id/index.php/jati/article/view/7152

Zhang, J. (2024). Impact of an improved random forest-based financial management model on the effectiveness of corporate sustainability decisions. *Systems and Soft Computing*, *6*, 200102. https://doi.org/10.1016/j.sasc.2024.200102

Zhu, Y., Tan, Y., Hua, Y., Wang, M., Zhang, G., & Zhang, J. (2010). Feature selection and performance evaluation of support vector machine (SVM)-based classifier for differentiating benign and malignant pulmonary nodules by computed tomography. *Journal of Digital Imaging*, *23*(1), 51–65. https://doi.org/10.1007/s10278-009-9185-9